Bioinformatics, 2017, 1–7
doi: 10.1093/bioinformatics/btx051
Advance Access Publication Date: 31 January 2017
Original Paper

OXFORD

Data and text mining

NSSRF: global network similarity search with subgraph signatures and its applications

Jiao Zhang, Sam Kwong, Yuheng Jia and Ka-Chun Wong*

Department of Computer Science, City University of Hong Kong, Kowloon Tong, Hong Kong

*To whom correspondence should be addressed.
Associate Editor: Jonathan Wren

Received on September 30, 2016; revised on December 9, 2016; editorial decision on January 20, 2017; accepted on January 24, 2017

Abstract

Motivation: The exponential growth of biological network database has increasingly rendered the global network similarity search (NSS) computationally intensive. Given a query network and a network database, it aims to find out the top similar networks in the database against the query network based on a topological similarity measure of interest. With the advent of big network data, the existing search methods may become unsuitable since some of them could render queries unsuccessful by returning empty answers or arbitrary query restrictions. Therefore, the design of NSS algorithm remains challenging under the dilemma between accuracy and efficiency.
Results: We propose a global NSS method based on regression, denotated as NSSRF, which boosts the search speed without any significant sacrifice in practical performance. As motivated from the nature, subgraph signatures are heavily involved. Two phases are proposed in NSSRF: offline model building phase and similarity query phase. In the offline model building phase, the subgraph signatures and cosine similarity scores are used for efficient random forest regression (RFR) model training. In the similarity query phase, the trained regression model is queried to return similar networks. We have extensively validated NSSRF on biological pathways and molecular structures; NSSRF demonstrates competitive performance over the state-of-the-arts. Remarkably, NSSRF works especially well for large networks, which indicates that the proposed approach can be promising in the era of big data. Case studies have proven the efficiencies and uniqueness of NSSRF which could be missed by the existing state-of-the-arts.
Availability and Implementation: The source code of two versions of NSSRF are freely available for downloading at https://github.com/zhangjiaobxy/nssrfBinary and https://github.com/zhangjiaobxy/nssrfPackage.
Contact: kc.w@cityu.edu.hk
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

High-throughput biotechniques have been applied to generate a significant amount of biological networks (Panni and Rombo, 2015). Examples include protein-protein interaction (PPI), protein-DNA interaction networks (Chatr-Aryamontri *et al.*, 2015), biological pathways (Kanehisa and Goto, 2000), transcription regulatory networks (Davidson *et al.*, 2002) and chemical compound structures (ChemIDplus) (Bank, 1998). With the exponentially increasing biological network data available, the bottleneck is no longer the lack of data but the way we are using it to explore new knowledge (Kalaev *et al.*, 2008). Comparative analyses of biological networks have as large impact as comparative genomics on our understanding of biology, species evolution, and disease detection (Sharan and Ideker, 2006). Such comparison could guide the transfer of knowledge across species and provide insights into the complex evolutionary history of protein interactions and protein functions based on the level of topological similarities among molecular networks (Faisal *et al.*, 2015; Xu *et al.*, 2015). The chemical structure search

© The Author 2017. Published by Oxford University Press. All rights reserved. For Permissions, please e-mail: journals.permissions@oup.com

technology and its query system, such as the National Library of Medicine (NLM) is widely used in biomedical fields (Bank, 1998). Figuring out a small subset of molecules for further analysis against the query structure can reduce the discovery cycles and human effort in drug design as well as other scientific activities (Willett *et al.*, 1998). In response, researchers have built various models from different views using sequence or topology information of networks for different purposes, which can be classified into two categories: network alignment (NA) and network similarity search (NSS).

NA aims to identify functional or topologically conserved components between two or among more networks; it indicates pairwise and multiple alignments, respectively (Sharan and Ideker, 2006). NA can be divided into two categories based on topological structure: local network alignment (LNA) and global network alignment (GNA). LNAs focus on finding highly conserved subgraphs without consideration of overall similarity. In contrast, GLAs focus on overall similarity maximization at the expense of suboptimal conservation in local regions. NA techniques provide metrics to evaluate how similar or dissimilar between the aligned networks. Comparative analysis of pathways can be used to transfer knowledge from well-studied species to poorly-studied species, thus yielding new insights into the structure of gene products which are important in evolutionary biology (Meng *et al.*, 2016).

NSS is a growing domain with widespread applications ranging from searching chemical compounds against a database of known molecules to matching evolutionarily conserved pathways/complexes that are structurally or functionally significant across species. Network search can be categorized as exact and inexact network search. (i) Exact network search requires one network being precisely a subgraph/graph of another network. However, exact network search is computationally infeasible due to NP-completeness of subgraph isomorphism problem (Cook, 1971; Döpmann, 2013). (ii) Inexact network search aims to find out networks that are similar to the query network either functionally or topologically (Raymond *et al.*, 2002). For instance, given a query network, the user may not know the exact overall structure he wants, while requires it to contain a particular set of functional fragments. NSS plays a key role by providing the best way to 'fit' a network into another network even the first network is not an exact subgraph/graph of the second one (Milenkovic *et al.*, 2010).

Given its significance, researchers have developed various techniques of NSS. GraphGrepSX (Bonnici *et al.*, 2010) is an indexing subgraph similarity search method based on suffix tree structure. NeMa is a neighborhood subgraph searching method (Khan *et al.*, 2013). MAGE is a pattern matching system which supports networks with both vertex and edge attributes based on the random walk with restart (RWR) algorithm (Pienta *et al.*, 2014). GString is a semantic-based approach which converts a network into the string representation, then the query is executed according to indices created in the preprocessing phase (Jiang *et al.*, 2007). However, all these works focused on the subgraph similarity search. Few works have been done on global NSS. C-tree is an indexing technique using K-NN query based on network edit distance, which returns K nearest networks against the query network (He and Singh, 2006). SIGMA is a set-cover-based NSS method, which is built on a variant of the set-cover problem (Mongiovi *et al.*, 2010). However, SIGMA fails to answer the query by returning empty answers in some cases. RINQ is a reference based index query method developed for searching biological pathway networks, which indexes the database by extracting a small set of networks as the reference to reduce the significant portion of the database (Gülsoy and Kahveci, 2011). REFBSS improved RINQ by defining a criterion for moving a portion of networks in the twilight zone to the result set without calculating similarity score costly (Soylev and Abul, 2015). However, both RINQ and REFBSS only work for small query networks with less than eight vertices, and the query time of these two methods increases exponentially with the growth of database, especially for the large networks.

We have developed a novel algorithm, Network Similarity Search based on Random Forest regression (NSSRF). It performs network search by considering the topology of query network and target network. Subgraph signatures of varying sizes and the cosine similarity score are taken into account. NSSRF has two phases: the offline model building phase and similarity query phase. In the offline model building phase, NSSRF utilizes subgraph signatures and cosine similarity score as features. The edge correctness (EC) and largest common connected subgraph (LCCS) generated by the pairwise NA method are considered as labels for efficient regression model training. In the similarity query phase, each query is inputted into the trained regression model from which the similarity score and similar networks are returned.

Our contribution is fourfold: (1) We solved the problem of 'query fails by returning empty answer or arbitrary query restrictions'. In this case, the top K similar networks in the database against each query network according to NA metric values are ranked and returned. (2) Existing NSS methods are mainly based on exact indexing, which can decrease the query time but result in low accuracy. NSSRF utilizes network topological feature of subgraph signatures for varying sizes, and it combines cosine similarity score between networks to construct the global features. Our model is based on random forest regression (RFR), which is still the state-of-the-arts (Fernández-Delgado *et al.*, 2014). (3) It is extremely time-consuming to search a large network database to figure out networks of interest. However, the query phase of NSSRF can be completed within seconds. (4) Existing NSS methods limit themselves to small networks, while NSSRF is scalable to large networks.

2 Materials and methods

2.1 Definitions and notations

2.1.1 Network and subgraph definition

A network is represented as a directed graph $N = (V, E)$ or N for brevity. N is constituted by a set of vertices $V(N)$ and edges $E(N)$, or V and E for brevity, respectively. The complete possible networks by a fixed number of vertices $|V|$ is $2^{|V|^2}$, which makes the analysis and modeling of network challenging. A network $N' = (V', E')$ is a subgraph of N, if there exists an isomorphism subgraph from N' to N, where $V' \subseteq V$ and $E' \subseteq E$. N' is called subgraph of N, reversely, N is the supergraph of N'; for instance, in biological networks, vertices are molecules and edges are interactions between molecules. Let $D = \{N_1, N_2, \ldots, N_n\}$ represents a network database, which contains n networks; N_q indicates a query network. The similarity search is to return all the networks which are topologically similar to the query network N_q.

2.1.2 Subgraph types and biological importance

Network motif is a special kind of subgraph occurring more frequently in the real network than those in randomized networks (Milo *et al.*, 2002). Different motifs indicate special structures and functions in the living cells; for instance, motifs in transcription networks carry out key functions, especially the feed-forward loop (FFL) (Mangan and Alon, 2003). The FFL is a three-gene subgraph that has two input transcription factors, one of which regulates the other, both jointly regulating a target gene. As motivated by the

nature that simple building blocks play key functions in biological networks, we utilize subgraphs of varying sizes to measure the similarity of network structure and function.

Subgraphs are extracted by detecting the subgraph isomorphism structure. MFinder (Kashtan *et al.*, 2004) can generate 2-node, 3-node and 4-node subgraphs within seconds for networks of less than 1000 vertices. Therefore, topological data of 2 to 4-node subgraphs are used as features in NSSRF model. There are 2, 13 and 199 kinds of 2-node, 3-node and 4-node subgraph, respectively. Let Sub_2N_i, Sub_3N_i, Sub_4N_i and $Sub_{234}N_i$ represent subgraph vectors of 2-node, 3-node, 4-node and the combination of these three subgraph types of network N_i, respectively.

2.1.3 Subgraph signatures similarity

NSSRF takes full advantage of network topology. Besides the subgraph frequency, similarity score between two networks is another feature used in NSSRF. The similarity can be estimated by Pearson correlation coefficient, cosine and Manhattan distance. Since the experimental results of subgraph frequency combined with each of these three metrics are very similar, we only use cosine similarity in NSSRF. Assuming $SubQ$ and $SubN$ are the vectors of k-node subgraph for network Q and N, respectively. Formula (1) is the cosine similarity between network Q and N of k-node subgraph. The example of normalized subgraph frequency and the cosine similarity score are shown in Supplementary Table S4. Take the cosine similarity of 2-node subgraph as an example. The total number of 2-node subgraph of the query network and the target network are 6 and 8. The frequency of subgraph ID2 of the query network and the target network are 5/6 and 1, and the frequency of subgraph ID6 of the query network and the target network are 1/6 and 0. Therefore, the cosine similarity between the query and target network is $\frac{\frac{5}{6}\times 1+\frac{1}{6}\times 0}{\sqrt{\left(\frac{5}{6}\right)^2+\left(\frac{1}{6}\right)^2}\sqrt{1^2+0^2}}=0.9806$.

$$\cos(Q,N)=\frac{\sum_{i=1}^{n} SubQ_i \cdot SubN_i}{\sqrt{\sum_{i=1}^{n}(SubQ_i)^2}\sqrt{\sum_{i=1}^{n}(SubN_i)^2}}. \quad (1)$$

2.1.4 Quality metrics of NA

To evaluate the quality of NA, researchers developed different criteria for LNA and GNA. In biology, LNA is measured biologically, such as functional consistency (FC) that evaluated the biological relations among vertices using gene ontology (GO) and human phenotype ontology (HPO) terms. GNA can be assessed both biologically and topologically (Meng *et al.*, 2016). NSSRF utilizes global structure features. Intuitively, a good NA algorithm should match vertices, conserve edges and detect the maximum commonly connected subgraph between aligned networks. The followings are widely used pairwise GNA quality metrics based on topology.

(1) EC is the percentage for edges in network N_i that are aligned to network N_j. The value of EC is in range of $[0, 1]$. For any two networks $N_i=(V_i,E_i)$ and $N_j=(V_j,E_j)$, the alignment of these two networks is an injective function f: $V(N_i)\rightarrow V(N_j)$ (Bunke and Shearer, 1998). Each edge $e=(u,v)\in E$ is associated with a score representing the bond strength. Formula (2) is the representation of EC (Milenkovic *et al.*, 2010):

$$EC=\frac{|\{(u,v)\in E_i : (f(u),f(v))\in E_j\}|}{|E_i|}. \quad (2)$$

(2) LCCS is the number of edges in the largest connected subgraph for the first network that is isomorphic to a subgraph of the second network (Hashemifar and Xu, 2014). For the aligned networks with the same EC score, one may prefer the network with large and contiguous regions of subgraphs rather than the one composed of many small and disconnected fragments. Supplementary Figure S1 illustrated the commonly connected subgraphs with different sizes. The value of LCCS is an integer in proportion to network size. In other words, the denser the network, the larger the LCCS is, and vice versa.

2.2 NSSRF approach

2.2.1 Random forest regression

Random forest (RF) is an ensemble learning method. A RF of N decision trees can be represented as $\{T_1(X),\ldots,T_N(X)\}$, where $X=\{x_1,\ldots,x_m\}$ is a m-dimensional feature vector containing normalized subgraph frequency and cosine similarity score. The ensemble method produces N decision tree outputs $\{\widetilde{Y_1}=T_1(X),\ldots,\widetilde{Y_N}=T_N(X)\}$, where $\widetilde{Y_i},(i=1,\ldots,N)$ is the predicted EC or LCCS value of the i-th tree. Outputs of all trees are aggregated to produce one final prediction $\widetilde{Y}$. For the regression problem in this study, $\widetilde{Y}$ is the average value of the N decisional tree predictions.

RF attempts to mitigate the problems of high variance and high bias by averaging them to find a natural balance between these two extremes. It combines tree predictors where each tree depends on the values of a random vector sampled independently with the same distribution for all trees in the forest. For each bootstrap sample, a tree is grown by choosing the best split among a randomly selected subset of maximum features rather than selecting all the features. We fix the tree number as 100, the number of maximum features is the only tuning parameter in the model by using root mean square error (RMSE) of Out-Of-Bag (OOB) samples. The basic principle is that a group of 'weak learners' can come together to form a 'strong learner'. In addition, RF has been proved not to be overfitting in theory (Breiman, 2001). Therefore, we have chosen RF for this regression task.

2.2.2 Framework overview

Network modeling provides a primitive and intuitive way for modeling biological data. The exponentially increasing network datasets render the query a computationally intensive work. Besides, existing techniques may not be able to answer the query by delivering an empty answer or adding arbitrary query restrictions. NSSRF has solved these limitations by returning a list of top similar networks according to their similarity score. Figure 1 visualizes the framework of NSSRF.

NSSRF has two phases: the first phase is offline model building, which has four steps in orange (F1 to F4). The basic idea of offline model building phase is extracting features from networks, then using labels to train a RFR model. Features used in NSSRF are the normalized frequency of varying sizes subgraph and cosine similarity score between the query and the target network. The similarity between a query and the target network is considered as NA problem. Therefore, we utilize NA technique to extract the alignment quality EC and LCCS value as labels. Herein, pairwise GNA method NETAL (Neyshabur *et al.*, 2013) and HubAlign (Hashemifar and Xu, 2014) are used to get alignment quality metrics. The second phase is similarity query, which has three steps in blue (S1 to S3). In this phase, the trained regression model is queried to return similar networks.

2.2.3 Network offline model training

Assuming $D=\{N_1,N_2,\ldots,N_n\}$ is the network database that has n networks. The network dataset used for model training is $T=\{T_1,T_2,\ldots,T_t\}$, which contains t networks. In Figure 1, the

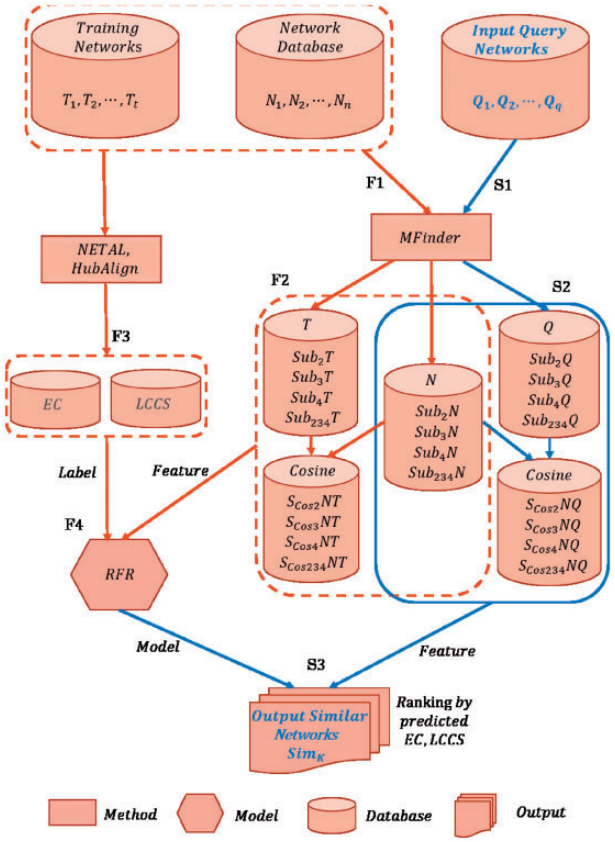

Fig. 1. Framework overview of NSSRF. The first phase is offline model training, which contains four steps (F1 to F4). F1 indicates step 1 of the first phase. The second phase is similarity query phase, which has three steps (S1 to S3). S1 indicates step 1 of the second phase

offline model training phase has four steps. The first step *F*1 is preparing input networks of training set T and database D, and MFinder (Kashtan *et al.*, 2004) is executed to extract subgraphs of varying sizes. This phase is computationally intensive, but it is fast for networks with edges less than 1000, and needs be done only once for the given network database. The feature vectors of 2-node, 3-node and 4-node subgraph can be represented as $Sub_2N = [Sub_2N_1, Sub_2N_2, \ldots, Sub_2N_n]^T$, $Sub_3N = [Sub_3N_1, Sub_3N_2, \ldots, Sub_3N_n]^T$ and $Sub_4N = [Sub_4N_1, Sub_4N_2, \ldots, Sub_4N_n]^T$, respectively. Therefore, the feature vector of combining these three subgraph types is $Sub_{234}N = [Sub_2N, Sub_3N, Sub_4N]$. The corresponding normalized subgraph frequency vectors are $Sub_2N_i = [f_{(i,1)}, f_{(i,2)}]$, $Sub_3N_i = [f_{(i,1)}, f_{(i,2)}, \ldots, f_{(i,13)}]$, $Sub_4N_i = [f_{(i,1)}, f_{(i,2)}, \ldots, f_{(i,199)}]$ and $Sub_{234}N_i = [Sub_2N_i, Sub_3N_i, Sub_4N_i]$, respectively.

After the subgraph normalization step *F*2, for each size of the subgraph, the cosine similarity between training network T and each network from the database D is calculated. The third step *F*3 is extracting the NA quality metric value EC and LCCS that are used as labels in the regression model. Supplementary Formula S(1) shows label vectors of EC and LCCS. Since our features used in NSSRF are topological data, we choose state-of-the-arts GNA techniques to validate the performance of NSSRF. The fourth step *F*4 is the RFR model building process. In RFR model, we use subgraph frequency Sub_kN, Sub_kT and cosine similarity score $S_{\cos k}NT$ as features, $L_{EC}NT$ and $L_{LCCS}NT$ are EC and LCCS labels. The RFR model will be used in the network similarity query phase. Supplementary Algorithm S1 outlines the pseudo-code for the offline model training of NSSRF. Note that we build a RFR model for each size of the subgraph to evaluate the full performance spectrum of NSSRF.

2.2.4 Network similarity query

The second phase is network similarity query, which contains three steps in Figure 1. The first two steps *S*1 and *S*2 are similar to the step *F*1 and *F*2 in the first phase. *S*1 is the subgraph frequency $Sub_k Q$ extraction step, and *S*2 is the cosine similarity $S_{\cos k}NQ$ calculation step between the query network and each network in the database.

In the third step *S*3, we use normalized subgraph frequency $Sub_k Q$ and cosine similarity score $S_{\cos k}NQ$ as features, then the RFR model is applied to predict the EC and LCCS of each pair (the query network and target network). Then, a list of the predicted alignment score of EC and LCCS are returned. Since the greater the EC and LCCS, the higher the similarity is, we chose to rank EC and LCCS in descending order. Therefore, the K most similar networks from the database against each query network are the top K ranked networks Sim_K. The performance of NSSRF for each size of the subgraph are examined. The pseudo-code for network similarity query of NSSRF is illustrated in Supplementary Algorithm 2.

3 Results

3.1 Datasets

To evaluate the performance of NSSRF, we have applied it to query biological pathway datasets and a molecular dataset. NSSRF is compared with C-tree, SIGMA and APPAGATO (Bonnici *et al.*, 2016) on the molecular dataset. Maximum vertex size of pathway dataset is 855, SIGMA returns empty answers for large network query in most cases. APPAGATO limits the number of vertex labels to no more than 256 and the query to be weakly connected. Therefore, unfortunately, NSSRF can only compared with C-tree on the pathway datasets.

The first dataset is the chemical structural data from NCI/NIH AIDS Antiviral Screen Data (http://dtp.cancer.gov). The other three datasets are pathway datasets downloaded from WikiPathways website (Kelder *et al.*, 2012) in March 2016. The datasets and network size are listed in Supplementary Table S1. The AIDS dataset contains the topological structures of chemical compounds that have been examined for anti-HIV activity. Each chemical compound is converted into a network such that vertices are atoms; edges are bonds between atoms. The multiple bonds are denoted by a single edge. 500 networks are used as the database and 30 networks are used as the query network set for 10-fold cross-validation in NSSRF.

In the WikiPathways dataset, since small networks are easy to be matched, we only consider networks with a degree more than three. Three trials on the pathway datasets of two species are conducted. In the first trial, Bos Taurus pathway dataset is examined, which has 230 networks. 30 networks are randomly selected as the query network set, and the other 200 networks are used as the database. Homo Sapiens pathway is used in the second and third trial. It contains 639 networks. In the second trial, 30 networks are also randomly selected as the query network set, and the other 609 networks are used as the database. In the third trial, the greatest 30 networks are chosen as the query networks, and the other 609 networks are used as the database. The reason we conduct the third trial is that dense network has complete coverage of the subgraph

types, which can fully demonstrate the search performance of NSSRF.

3.2 Topological quality metric of NSS

To guarantee the experiment quality, we use 10-fold cross-validation for performance comparison. EC and LCCS are utilized as network similarity criteria. Specifically, we have measured the overlap of Jaccard similarity coefficient (Jaccard, 1901), which is defined in formula (3):

$$Overlap = \frac{|O_A \cap O_S|}{|O_A \cup O_S|}, \tag{3}$$

where, O_A and O_S indicate top K similar networks detected by NA and NSS technologies, respectively. The maximum value of overlap is 1 when the top K similar networks identified by NA and NSS methods are entirely overlapped.

When we evaluate the quality of similarity query on EC and LCCS generated by NETAL (Neyshabur *et al.*, 2013), the overlap performance comparison of Jaccard similarity coefficient on the four real world datasets is shown in Figure 2. For AIDS dataset, the overlap performance of NSSRF using 2-node subgraph is very close to SIGMA, C-tree and APPAGATO. Using 3-node, 4-node and the combination of 2-, 3-, 4-node subgraph, the overlap performance of NSSRF are significantly better than SIGMA, C-tree and APPAGATO. For the three WikiPathways datasets, SIGMA and APPAGATO do not support such a large dataset, we only compare NSSRF with C-tree. The overlap performance of NSSRF for 2-node subgraph evaluated by EC performs better than C-tree, while performs very close to C-tree for LCCS. NSSRF using 3-node, 4-node and the combination of 2-, 3-, 4-node subgraph perform significantly better than C-tree in most cases. The difference between Homo Sapiens I and II dataset is that the training query networks of Homo Sapiens I dataset are randomly selected, while the training query networks of Homo Sapiens II dataset are the greatest networks in the database. For the greatest 30 networks under 10-fold cross-validation, the overlap performance is much better than randomly picked networks since large network covers complete subgraph information that can be fully utilized. Moreover, when HubAlign (Hashemifar and Xu, 2014) is used to judge the query quality, the overlap performance on the four real world datasets is very similar to NETAL, which is shown in Supplementary Figure S4.

EC and LCCS scores returned by the C-tree, SIGMA, APPAGATO and NSSRF on the four datasets are depicted in Supplementary Figures S6 and S7. The statistical significance using *t*-test and Wilcoxon's rank-sum test on EC and LCCS are computed and ascertained on the returned results. For the AIDS dataset, NSSRF using 2-node subgraph is limited in performance. However, NSSRF using other subgraph settings can show better query performance than C-tree, SIGMA and APPAGATO in terms of EC and LCCS. For the WikiPathways datasets, NSSRF can outperform C-tree (the only runnable program) significantly in most cases.

3.3 Biological quality metric of NSS

Functional coherence reflects biological significance by measuring the semantic consistency of GO terms and HPO terms assigned to the corresponding protein-coding genes. GO annotations for all the involved proteins are extracted from the Gene Ontology Consortium (Ashburner *et al.*, 2000), and HPO annotations are extracted from Human Phenotype Ontology database (Köhler *et al.*, 2016). The similarity between two pathways is computed using the biological function distance, i.e. the smaller the distance, the more similar the two pathways are. Canberra distance (Lance and Williams, 1967) is adopted to calculate the biological function distance, which is defined in formula (4):

$$Distance = \sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|Q_i - N_j|}{|Q_i| + |N_j|}, \tag{4}$$

where, Q and N are the query network and the target network; there are m and n proteins in Q and N respectively. Q_i and N_j are the binary vectors denoting the presence of GO terms of the ith protein of Q and jth protein of N, respectively. Such distance calculation strategy is also adopted for the HPO terms. Since the AIDS dataset is limited to the chemical compound that does not have any GO or HPO term, we focus on the three WikiPathways datasets on which C-tree can be executed. For each given query network, the Canberra distances of the top 10% similar networks returned by C-tree and NSSRF to each query network are calculated for performance comparisons. Both t-test and Wilcoxon's rank-sum test are computed to ascertain the statistical significances ($P \leq 0.01$) which are shown in Supplementary Figures S5 and S9. NSSRF marked with asterisk denotes that the performance of NSSRF is better than other methods with statistical significance. Since the statistical significance test result of t-test and Wilcoxon's rank-sum test are very similar in this study, only the significance result of Wilcoxon's rank-sum test on the corresponding figures is marked. It is observed that the Canberra distances between the query network and the returned networks in terms of GO and HPO terms by NSSRF are smaller than C-tree in most cases. It gives additional proofs that NSSRF can return biologically relevant results which are not limited to the topological overlap metric.

3.4 Speed

Two phases are involved in NSSRF, the offline model training phase and network similarity query phase. SIGMA and C-tree also have two phases, we compare their time cost with NSSRF on offline model training phase and similarity query phase, respectively. Since APPAGATO does not have offline model training phase, we only compare NSSRF with APPAGATO on the similarity query phase. Runtime of selecting top 10% similar networks is collected. The experiment was done on Ubuntu with Intel(R) Core(TM) i7-4790 processor and 8G memory.

The time of offline model training and similarity query of NSSRF and the comparison methods on AIDS and WikiPathways datasets are shown in Supplementary Figures S8 and S10, respectively. Since the average query time of SIGMA is approximately 50 thousand times of other methods, we show the runtime in $\log 1p$ scale on the figures. The total runtime of NSSRF for 2-node subgraph is the smallest. The training time of NSSRF for 3-node, 4-node and the combination of 2-, 3-, 4-node subgraph are comparable with C-tree and SIGMA. In addition, the query time of NSSRF is the smallest.

3.5 Space and time complexity

The practical memory usage of NSSRF including feature extraction and offline model training on the four real world datasets are tabulated in Supplementary Table S2. The practical memory usage on the four datasets is approximately 1GB in Linux. The time complexity of feature extraction using Mfinder is $S_T * O(C^{k-1} * k^{k+1}))$, where S_T is the number of samples for the sampling method to detect subgraphs, C is a small constant correlated with the average degree of nodes in the network, and k is the subgraph node size. The space complexity of feature extraction is $O(k^2 + E)$, E is the number of

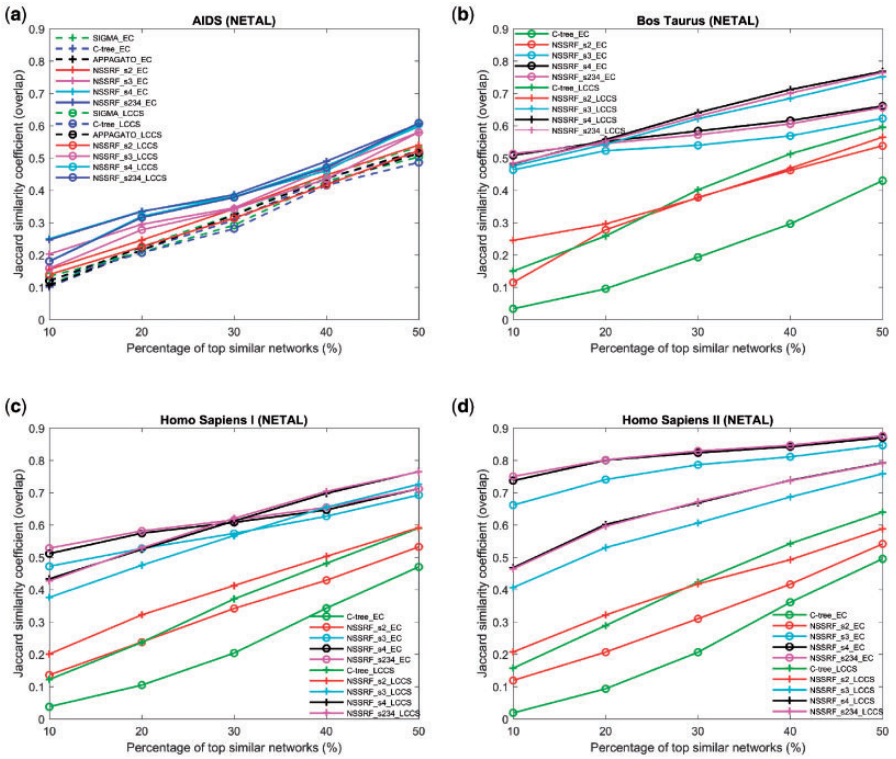

Fig. 2. The overlap performance comparison of Jaccard similarity coefficient on AIDS, Bos Taurus, Homo Sapiens I and Homo Sapiens II datasets under 10-fold cross-validation. EC and LCCS generated by NETAL is used. _EC and _LCCS indicate evaluating the performance of NSSRF on EC and LCCS, respectively. NSSRF_s2_EC indicate evaluating NSSRF with 2-node subgraph on EC, which is the same as the other subgraph sizes used in NSSRF. Note: Only C-tree and NSSRF can be run on the WikiPathways datasets

edges in the network (Kashtan *et al.*, 2004). In RFR model training phase, assuming each decision tree is a binary tree; the number of instances is n, and the average depth of tree is $\log(n)$; the number of variables to sample at each node is m, time complexity to build one tree is $O(m * n\log(n))$. Therefore, the total time complexity to build a RF of t trees is $O(t * m * n\log(n))$; the space complexity is $O(t * \sum_{i=0}^{\log(n)-1} 2^i)$, where $\sum_{i=0}^{\log(n)-1} 2^i$ is the number of nodes in each of the t decision trees.

3.6 Case studies

Two case studies are conducted to verify the performance of NSSRF. The query networks are randomly selected. The first case study is derived from the AIDS dataset in which the frequencies of 4-node subgraph signature are used as features in NSSRF. The ECs generated by NETAL are adopted as target labels in NSSRF in the RFR offline model training phase. In the query phase, a query chemical compound #502 (Supplementary Fig. S3(a)) out of the database is randomly picked to query the AIDS database with 500 networks. A chemical compound #100 (Supplementary Fig. S3(b)) in the database is also randomly picked. We use NETAL to extract the EC values between #502 and networks in the AIDS database, then ranking the networks in the database in descending order. We found that EC between #502 and #100 is 0.88, and #100 is located in the top 3% similar networks. In addition, the EC between #502 and the most similar network in the database is 0.93. When we use NSSRF to query the database and obtain the top 5% similar networks, NSSRF can hit well the target #100. However, SIGMA, C-tree and APPAGATO fail to hit the target even in their top 10% similar networks.

The second case study is conducted on the WikiPathways Bos Taurus dataset. The frequencies of 3-node subgraph signature are used as features. The LCCSs generated by NETAL are used as target labels in the RFR offline model training phase. In the query phase, a query pathway (i.e. IL-1 signaling pathway WP3271 (Supplementary Fig. S2(a))) out of the database is randomly selected to query the Bos Taurus dataset with 200 pathways. We use NETAL to calculate the LCCS values between pathway WP3271 and pathways in the Bos Taurus dataset, the pathways are sorted in descending order. We found that LCCS between WP3271 and a pathway in the database (i.e. ATM signaling pathway WP3221 (Supplementary Fig. S2(b))) is 16, and WP3221 is located in the top 10% similar pathways. The Canberra distance in terms of GO and HPO terms between the WP3271 and WP3221 are 31 and 16, respectively. Furthermore, the LCCS, Canberra distance in terms of GO and HPO between WP3271 and the most similar pathway in the database are 24, 12 and 10, respectively. When NSSRF is used to obtain the top 10% similar pathways, it can hit the WP3221. In contrast, unfortunately, C-tree fails to hit WP3221 even in its top 30% similar pathways, while SIGMA and APPAGATO cannot be run on those networks to the best of our efforts.

4 Discussion

In this work, we have developed NSSRF to find out similar networks from a given database. There are two phases in NSSRF: offline model training phase and network similarity query phase.

The overlap performance of NSSRF on 2-node subgraph outperforms SIGMA, C-tree and APPAGATO in most cases. NSSRF using other subgraph settings can perform significantly better than the other three methods. Unfortunately, SIGMA and APPAGATO are found not scalable to large networks. APPAGATO also requires the query network to be weakly connected. In contrast, NSSRF does not have such restrictions even for large networks. In addition, NSSRF outperforms C-tree significantly in terms of the functional consistency of GO and HPO terms. Besides, we have observed that the cosine similarity between subgraphs is an important feature in the network topology search performance. Moreover, NSSRF is built on RF which can be protected from model overfitting.

The offline model training time of NSSRF is comparable to SIGMA and C-tree. However, the query speed of NSSRF is much faster than the other three methods. In order to investigate the limitations of NSSRF, five PPI networks of varying sizes are examined under the NSSRF methodology as shown in the Supplementary Table S3. The results indicate that the subgraph extraction is the limiting step in NSSRF due to the NP-completeness of subgraph isomorphism.

Acknowledgements

The authors would like to thank the reviewers for their constructive comments which have improved the study in numerous ways. The authors would also like to thank the authors of SIGMA, C-tree and APPAGATO for making their programs available. In addition, the authors would also like to thank the AIDS Antiviral Screen and WikiPathways community for making their data available.

Funding

The work described in this paper was partially supported by a grant from City University of Hong Kong (CityU Project No. 72004444) and an Early Career Scheme grant from Research Grant Council (CityU Project No. 9048072 and RGC Project No. 21200816) in Hong Kong.

Conflict of Interest: none declared.

References

Ashburner,M. *et al*. (2000) Gene ontology: tool for the unification of biology. *Nat. Genet*., **25**, 25–29.

Bank,H.S.D. (1998) National library of medicine. *Bethesda, Maryland (TOMES CPS# CD-ROM)*. https://www.nlm.nih.gov/pubs/factsheets/hsdbfs.html.

Bonnici,V. *et al*. (2010). Enhancing graph database indexing by suffix tree structure. In: *IAPR International Conference on Pattern Recognition in Bioinformatics*, pp. 195–203. Springer.

Bonnici,V. *et al*. (2016) APPAGATO: an APproximate PArallel and stochastic GrAph querying TOol for biological networks. *Bioinformatics*, **32**, 2159–2166.

Breiman,L. (2001) Random forests. *Mach. Learn*., **45**, 5–32.

Bunke,H. and Shearer,K. (1998) A graph distance metric based on the maximal common subgraph. *Patt. Recogn. Lett*., **19**, 255–259.

Chatr-Aryamontri,A. *et al*. (2015) The biogrid interaction database: 2015 update. *Nucleic Acids Res*., **43**, D470–D478.

Cook,S.A. (1971). The complexity of theorem-proving procedures. In: *Proceedings of the third annual ACM symposium on Theory of computing*, pp. 151–158. ACM.

Davidson,E.H. *et al*. (2002) A genomic regulatory network for development. *Science*, **295**, 1669–1678.

Döpmann,C. (2013) Survey on the graph alignment problem and a benchmark of suitable algorithms. Bachelor's Thesis, *Institut Für Informatik*, Humboldt-Universitat zu Berlin.

Faisal,F.E. *et al*. (2015) The post-genomic era of biological network alignment. *EURASIP J. Bioinf. Syst. Biol*., **2015**, 1.

Fernández-Delgado,M. *et al*. (2014) Do we need hundreds of classifiers to solve real world classification problems. *J. Mach. Learn. Res*., **15**, 3133–3181.

Gülsoy,G. and Kahveci,T. (2011) Rinq: Reference-based indexing for network queries. *Bioinformatics*, **27**, i149–i158.

Hashemifar,S. and Xu,J. (2014) Hubalign: an accurate and efficient method for global alignment of protein–protein interaction networks. *Bioinformatics*, **30**, i438–i444.

He,H. and Singh,A.K. (2006). Closure-tree: An index structure for graph queries. In: *22nd International Conference on Data Engineering (ICDE'06)*, pp. 38–38. IEEE.

Jaccard,P. (1901) A comparative study of the floral distribution in alps and jura. *Bull. Walden Soc. Nat. Sci*., **37**, 547–579.

Jiang,H. *et al*. (2007). Gstring: A novel approach for efficient search in graph databases. In: *2007 IEEE 23rd International Conference on Data Engineering*, pp. 566–575. IEEE.

Kalaev,M. *et al*. (2008) Networkblast: comparative analysis of protein networks. *Bioinformatics*, **24**, 594–596.

Kanehisa,M. and Goto,S. (2000) Kegg: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res*., **28**, 27–30.

Kashtan,N. *et al*. (2004) Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. *Bioinformatics*, **20**, 1746–1758.

Kelder,T. *et al*. (2012) Wikipathways: building research communities on biological pathways. *Nucleic Acids Res*., **40**, D1301–D1307.

Khan,A. *et al*. (2013). Nema: Fast graph search with label similarity. In: *Proceedings of the VLDB Endowment*, vol. **6**, pp. 181–192. VLDB Endowment.

Köhler,S. *et al*. (2016) The Human Phenotype Ontology in 2017. *Nucleic Acids Res*., gkw1039.

Lance,G.N. and Williams,W.T. (1967) Mixed-data classificatory programs i – agglomerative systems. *Aust. Comput. J*., **1**, 15–20.

Mangan,S. and Alon,U. (2003) Structure and function of the feed-forward loop network motif. *Proc. Natl. Acad. Sci*., **100**, 11980–11985.

Meng,L. *et al*. (2016) Local versus global biological network alignment. *Bioinformatics*, **32**, 3155–3164.

Milenkovic,T. *et al*. (2010) Optimal network alignment with graphlet degree vectors. *Cancer Inf*., **9**, 121.

Milo,R. *et al*. (2002) Network motifs: simple building blocks of complex networks. *Science*, **298**, 824–827.

Mongiovi,M. *et al*. (2010) Sigma: a set-cover-based inexact graph matching algorithm. *J. Bioinf. Comput. Biol*., **8**, 199–218.

Neyshabur,B. *et al*. (2013) Netal: a new graph-based method for global alignment of protein–protein interaction networks. *Bioinformatics*, **29**, 1654–1662.

Panni,S. and Rombo,S.E. (2015) Searching for repetitions in biological networks: methods, resources and tools. *Brief. Bioinf*., **16**, 118–136.

Pienta,R. *et al*. (2014). Mage: Matching approximate patterns in richly-attributed graphs. In: *2014 IEEE International Conference on Big Data (Big Data)*, pp. 585–590. IEEE.

Raymond,J.W. *et al*. (2002) Rascal: Calculation of graph similarity using maximum common edge subgraphs. *Comput. J*., **45**, 631–644.

Sharan,R. and Ideker,T. (2006) Modeling cellular machinery through biological network comparison. *Nat. Biotechnol*., **24**, 427–433.

Soylev,A. and Abul,O. (2015). Refbss: Reference based similarity search in biological network databases. In: *2015 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pp. 1–8. IEEE.

Willett,P. *et al*. (1998) Chemical similarity searching. *J. Chem. Inf. Comput. Sci*., **38**, 983–996.

Xu,K. *et al*. (2015) Genomic and network patterns of schizophrenia genetic variation in human evolutionary accelerated regions. *Mol. Biol.Evol*., msv031.